COMP642: Machine Learning

- **Instructor:** Anshumali Shrivastava (anshumali AT rice.edu)
- **Class Timings:** Tuesday and Thursday 1pm - 2:15pm
- **Location:** HRZ 212
- **Teaching Assistants:** TBA
- **Office Hours:** TBA

Structure

Description: This class aims to train future professional industry leaders in machine learning. The course will focus on the foundation and practice of widely adopted modern ideas and principles that make a difference in real applications.

Redesigned Topics: We are well aware that many well-established ideas in traditional machine learning are either becoming obsolete or are getting questioned under the light of more observations and experimentations with deep learning. This course is redesigned to eliminate them. Most existing mathematics fails to explain the success of deep and machine learning. Also, most successful models such as transformers, etc., came independent of the rigorous understanding of deep learning. A primary aim of this course is to develop an instinct for practical machine learning via case studies and assignments, which is one of the essential skills for success in the field.

Coverage: We will cover all aspects of modern machine learning (See schedule below), including Deep Learning Architectures, Graph Neural Networks, Self Supervised Learning, Tiny ML, Distributed and Federated ML, etc. The course will also demonstrate that while machine learning seems to have too many topics, the motivating fundamentals are only a handful.

Prerequisite: A prior coursework on machine learning is preferred but not required. It turns out that the most sophisticated machine learning systems and algorithms of today do not require significant mathematical preparations. Basic probability and multivariate calculus, along with Linear algebra at the vector spaces and matrix manipulations, are sufficient mathematical foundations for this course. The course **does require** rigorous experience in programming. Design and analysis of algorithms and basic data structures to understand compute and memory efficiency. Basic knowledge of computer systems such as cache hierarchy, memory latency will be required to understand the practicality of ideas used in modern ML systems.

If you are unsure about your prerequisites, please contact the instructor.

Materials: Machine Learning is moving very fast. Techniques published a few years back are getting obsolete and so are textbooks and most known courses. There are no single textbooks for the materials covered in this class, some of the materials are ideas that appeared last year (Such as MLP Mixer). Lecture recordings and references will be provided during the class and will be available on the course page. Having said that, any good tutorial on deep learning with Python will be helpful for exercises.

Grading

- Term Project: 25%
- 4-5 Assignments: 35%
- 2 Quizzes: 30%
- Lecture Scribing: 10%

Schedule

- **1/11: Machine Learning as a new paradigm for programming.**
  - Logistics, Course Content, Difference between AI Course and This Machine Learning Course
  - Notion of Supervised Data. Concept of: Training, Inference, and Models.
  - Programming with Data. A new human-like approach to solve problems.
  - Discussion: Best Known Sorting Algorithms Vs Sorting the Machine Learning way! The case of learned sorting that outperforms all sorting algorithms! (Essentially the Magic of Machine Learning: Seemingly Hopeless things can be guessed by ML Process reasonably well!)
- **1/13: Opening Up the Black Box: Supervised Learning Algorithms (a.k.a. Descent on Input-Output Pairs)**
  - Machine Learning as Optimization: Function Class, Loss Function or Objective
  - The basic assumption of "Learnability"
  - Iterative Updates: Move a step closer. Notions of Descent Direction and Simple Contraction based Proof. You don't need much to prove convergence!
  - Noisy Gradient Descents are equally good (Sometimes better)!
  - Oscillations and The Unsolved Challenge of Optimal Step Size: Cannot afford to compute it.
  - Problem of fixed step size and Ideas. Normalizing gradients with mean and standard deviation leading to Adam and other adaptive variants.
  - Running Average, decaying average, velocity, (Efficient Proxies for mean and standard deviations
- **1/18: Linear Classifiers I**
  - Restrict the model to be linear.
  - Perceptron as Linear Models. Logistic Regression (A very popular classifier)
  - Over Fitting with limited data and Constraining the models (Regularizations)
  - Large-Margin SVMs as regularized hinge loss.
- **1/20: Linear Classifiers II**
  - Issues with Linear Classifiers, XOR problem, (XOR problem demonstrates fundamentally that there is really no substitute for Feature Engineering! Sort of Information Paradox for learning: Equal information content in features does not imply similar learning outcomes.)
  - Kernels (Non-Linear Learning) and Feature Expansions.
  - Overparameterization: The phenomena of Double Descent
- **1/25: Deep Learning: Logistic Regression with Sequential Non-Linear Feature Extraction catered to minimize the loss.**
  - Multi Layer Perceptron and Universal Approximators
  - Why Deep Learning is Feature Learning? The notion of embeddings and deeper representations.
  - Understanding The Information bottleneck.
  - Densely connected variants and Concatenation Tricks to overcome information bottleneck
- **1/27: Gradient Descent with Chain Rule Aka Backpropagation. Matrix Multiplications and GPUs.**
  - The Chain Rule and Classical Backpropagation.
  - Batch Gradient Descent as matrix multiplication.
  - Simple Tricks utilized by popular softwares to implement batch matrix multiplication on GPUs. Issues with replicated memory and scaling for large matrices.
- **2/1:Decision Trees**
  - Notion of Decision Trees
  - Classification and Regression Trees (CART)
- **2/3:Near-Neighbor Search**
  - Near Neighbor Classification.
  - The Good old Problem of Near-Neighbor Search and classical algorithms.
  - A preview of the of recent advances in efficient search based graph based near-neighbor search and learning heuristics

- **2/8: Ensembles: Bagging and Boosting, XGBoost (Another Popular Classifier)**
  - Bagging
  - Boosting: XGBoost
  - Case Study of Amazon's AutoGluon
- **2/8: SPRING BREAK**
- **2/15: Learning with Data Types 1: Sets, Text, n-grams, Documents with Embedding Models**
  - Bag of Words
  - Embeddings for Tokens: Generic Featurization for Text, Tokens and Categories.
- **2/17: Learning with Data Types 2: Images and Time Series with Convolutional Features**
  - Featurization of Images and Time Series using Convolutions and CNNs
  - Old Tokenization, BoW for Images. Patches are almost words!
  - MLP Mixers: Images are not different now.
  - Count Features for Time Series.
- **2/22: Learning with Data Types 3: Categorical and Tabular Data: Deep Embedding Models, Pairwise Kernels and Cross Correlation: Attention**
  - Popular Deep Embedding Model, Embedding Tables for Efficiency.
  - Good Old Pairwise Correlations, A Popular twist goes by the name "attention".
  - A Note for Trees and Boosted Trees for Categorial Variables.
- **2/24: Learning with Data Types 4: Graphs and relations**
  - Page Rank, Learned Weighted Pagerank Vectors a.k.a Graph Neural Networks (GCNN)
- **3/1: Semi-Supervised Learning**
  - How can unlabelled data help?
  - Co-training
  - Ladder Networks, Reconstruction Loss, and more.
- **3/3: Unsupervised Learning**
  - Autoencoders, Clustering, Embeddings
- **3/8: Self-Supervised Learning**
  - Case Study: GPT-3 and pre-trained embeddings in NLP
  - Auxiliary Tasks and Labels.
- **3/10: Generative Models and GANs**
  - Implicit Soft Contraints: Constrain your network with another network.
  - Training Challenges
  - An overview of results.
- **3/15: SPRING BREAK**
- **3/17: SPRING BREAK**
- **3/22: ML for Production and Deployment: Large Scale ML - 1**
  - Subsampling, The Hashing Trick, Sparsity.
- **3/24: ML for Production and Deployment: Large Scale ML - 2**
  - Caching Results, Pre-Training, Embedding Tables, etc.
- **3/29: Distributed ML - 1**
  - Data Parallel and Model Parallel Training.

- **3/31: Distributed ML - 2**
  - Hardware Friendly Model and Gradient Compressions: Bloom Filters, Count Sketch, etc.
- **4/5: Generic Neural Information Retrieval and Learning with Large Output Spaces**
  - Negative Sampling, Question Answering and Translation (Embedding and Lookups).
  - A Case study of Modern Search and Recommendation Systems.
- **4/7: ML on Edge and IoT: Tiny ML- 1:**
  - Quantizations and Pruning.
- **4/12: ML on Edge and IoT: Tiny ML- 2:**
  - Knowledge Distillation.
- **4/14: Slack Class or Special Topics:**

Students with Disability

If you have a documented disability that may affect academic performance, you should: 1) make sure this documentation is on file with Disability Support Services (Allen Center, Room 111 / adarice@rice.edu / x5841) to determine the accommodations you need; and 2) meet with me to discuss your accommodation needs.